**Dockerfile**

FROM golang:1.23-alpine

WORKDIR /app

COPY go.mod go.sum ./

RUN go mod download

COPY . .

RUN go mod tidy && go build -o main .

EXPOSE 8080

CMD ["./main"]

В рабочую директорию контейнера app копируются файлы go.mod go.sum (отвечают за зависимости), запускается загрузка зависимостей из go mod, копируются файлы из текущей папки в папку контейнера. Приводим в порядок зависимости через go mod tidy и комплириуем приложение через go build -o main.

**.env**

PORT=4000

DEBUG=true

ALLOWED\_HOSTS=0.0.0.0

ALLOWED\_ORIGINS=\*

SERVER\_TIMEZONE=Asia/Kolkata

DATABASE\_URI=mongodb://mongo:27017

DATABASE\_NAME=gbs\_db

JWT\_SECRET=gbs\_secret

JWT\_EXPIRY\_HOURS=14

**Соберем образ**:

docker build -t gin-backend .

**docker-compose**

version: '3.8'

services:

gbs:

build:

context: .

dockerfile: Dockerfile

ports:

- "4000:4000"

env\_file:

- .env

depends\_on:

- mongo

restart: unless-stopped

mongo:

image: mongo:6

ports:

- "27017:27017"

volumes:

- mongo\_data:/data/db

environment:

MONGO\_INITDB\_ROOT\_USERNAME: root

MONGO\_INITDB\_ROOT\_PASSWORD: root

MONGO\_INITDB\_DATABASE: gbs\_db

restart: unless-stopped

blackbox:

image: prom/blackbox-exporter

ports:

- "9115:9115"

volumes:

- ./blackbox.yml:/etc/blackbox\_exporter/config.yml

restart: unless-stopped

prometheus:

image: prom/prometheus

ports:

- "9090:9090"

volumes:

- ./prometheus.yml:/etc/prometheus/prometheus.yml

restart: unless-stopped

volumes:

mongo\_data:

**prometheus.yml**

global:

scrape\_interval: 30s

scrape\_configs:

- job\_name: "blackbox\_http\_checks"

metrics\_path: /probe

params:

module: [http\_2xx]

static\_configs:

- targets:

- http://gbs:4000/health

relabel\_configs:

- source\_labels: [\_\_address\_\_]

target\_label: \_\_param\_target

- source\_labels: [\_\_param\_target]

target\_label: instance

- target\_label: \_\_address\_\_

replacement: blackbox:9115

**blackbox.yml**

modules:

http\_2xx:

prober: http

timeout: 5s

http:

method: GET

valid\_http\_versions: ["HTTP/1.1"]

valid\_status\_codes: [200]

**Запустим всё**:

docker compose up -d